

CLAIMS

What is claimed is:

- 1 1. A method for transparency rendering in a graphics pipeline, comprising:
 - 2 (a) collecting colored-transparency information from a plurality of depth layers
 - 3 in a scene to be rendered;
 - 4 (b) storing the collected colored-transparency information in memory; and
 - 5 (c) blending the colored-transparency information from the depth layers in a
 - 6 predetermined order.
- 1 2. The method as recited in claim 1, wherein the colored-transparency
- 2 information is collected from at least two depth layers.
- 1 3. The method as recited in claim 1, wherein the colored-transparency
- 2 information is stored in a plurality of texture maps.
- 1 4. The method as recited in claim 3, wherein each of the texture maps
- 2 corresponds with one of the depth layers.
- 1 5. The method as recited in claim 4, wherein the texture maps are stored in
- 2 memory.
- 1 6. The method as recited in claim 1, and further comprising rendering opaque
- 2 objects in the scene.
- 1 7. The method as recited in claim 6, the opaque objects in the scene are
- 2 rendered prior to blending the colored-transparency information therewith.
- 1 8. The method as recited in claim 1, wherein the memory includes a frame
- 2 buffer.

- 1 9. The method as recited in claim 1, wherein the blending includes linear
2 blending.
- 1 10. The method as recited in claim 1, wherein the colored-transparency
2 information is collected utilizing depth peeling.
- 1 11. The method as recited in claim 10, wherein the depth peeling includes
2 executing a first rendering pass for collecting colored-transparency
3 information relating to a first depth layer, and executing additional rendering
4 passes for collecting additional colored-transparency information relating to
5 additional depth layers.
- 1 12. The method as recited in claim 11, wherein the first rendering pass produces
2 a shadow map relating to the first depth layer.
- 1 13. The method as recited in claim 11, wherein a shadow-mapping feature is
2 enabled during the additional rendering passes for defining a previous depth
3 layer.
- 1 14. The method as recited in claim 11, wherein the additional rendering passes
2 are taken from the same eye position from which the first rendering pass is
3 taken.
- 1 15. The method as recited in claim 1, wherein the colored-transparency
2 information is collected utilizing depth peeling including executing a first
3 rendering pass for generating a shadow map from which first colored-
4 transparency information relating to a first depth layer is collected, and
5 executing additional rendering passes with a shadow-mapping feature
6 enabled and from the same eye position from which the first rendering pass is

7 taken for collecting additional colored-transparency information relating to
8 additional depth layers.

1 16. The method as recited in claim 15, wherein the additional colored-
2 transparency information relating to the additional depth layers is collected
3 by removing a portion of the scene associated with a previous depth layer.

1 17. The method as recited in claim 16, wherein the additional colored-
2 transparency information relating to the additional depth layers is collected
3 by performing a test to determine which portion of the scene to remove.

1 18. The method as recited in claim 17, wherein the test determines whether the
2 portion of the scene is behind the previous depth layer.

1 19. The method as recited in claim 18, wherein the portion of the scene is
2 removed upon the test determining that the portion of the scene is behind the
3 previous depth layer.

1 20. The method as recited in claim 19, wherein the test calculates a difference
2 between a previous z-value relating to the previous depth layer and a present
3 z-value relating to one of the additional depth layers.

1 21. The method as recited in claim 20, wherein the portion of the scene is
2 removed upon no difference being calculated between the previous z-value
3 relating to the previous depth layer and the present z-value relating to one of
4 the additional depth layers.

1 22. The method as recited in claim 21, wherein the z-values relating to all depth
2 layers are produced with the same interpolation-related method for
3 improving an accuracy of the test.

- 1 23. A computer program product for transparency rendering in a graphics
2 pipeline, comprising:
3 (a) computer code for collecting colored-transparency information from a
4 plurality of depth layers in a scene to be rendered;
5 (b) computer code for storing the collected colored-transparency information in
6 memory; and
7 (c) computer code for blending the colored-transparency information from the
8 depth layers in a predetermined order.
- 1 24. A system for transparency rendering in a graphics pipeline, comprising:
2 (a) logic for collecting colored-transparency information from a plurality of
3 depth layers in a scene to be rendered;
4 (b) memory for storing the collected colored-transparency information; and
5 (c) a renderer coupled to the memory for blending the colored-transparency
6 information from the depth layers in a predetermined order.
- 1 25. A system for transparency rendering in a graphics pipeline, comprising:
2 (a) logic for collecting colored-transparency information from a plurality of
3 depth layers in a scene to be rendered;
4 (b) memory for storing the collected colored-transparency information; and
5 (c) register combiners coupled to the memory for blending the colored-
6 transparency information from the depth layers in a predetermined order.
- 1 26. A method for transparency rendering in a graphics pipeline, comprising:
2 (a) collecting colored-transparency information from at least two depth layers in
3 a scene;
4 (b) storing the collected colored-transparency information in the form of a
5 plurality of texture maps;
6 (c) rendering the opaque objects in the scene;
7 (d) storing the rendering of the opaque objects in memory;
8 (e) identifying one of the depth layers to be blended;

- 9 (f) blending the colored-transparency information from the identified depth layer
10 with contents of the memory utilizing a corresponding one of the texture
11 maps;
12 (g) storing results of (f) in the memory; and
13 (h) repeating acts (e)-(g).

- 1 27. A computer program product for transparency rendering in a graphics
2 pipeline, comprising:
3 (a) computer code for collecting colored-transparency information from at least
4 two depth layers in a scene;
5 (b) computer code for storing the collected colored-transparency information in
6 the form of a plurality of texture maps;
7 (c) computer code for rendering opaque objects in the scene;
8 (d) computer code for storing the opaque object in memory;
9 (e) computer code for identifying one of the depth layers to be blended;
10 (f) computer code for blending the colored-transparency information from the
11 identified depth layer with contents of the memory utilizing a corresponding
12 one of the texture maps;
13 (g) computer code for storing results of (f) in the memory; and
14 (h) computer code for repeating acts (e)-(g).

- 1 28. A method for transparency rendering in a graphics pipeline, comprising:
2 (a) collecting colored-transparency information from a plurality of depth layers
3 in a scene to be rendered by:
4 (i) executing a first rendering pass for generating a shadow map and for
5 collecting first colored-transparency information relating to a first depth
6 layer, and
7 (ii) executing additional rendering passes with a shadow-mapping feature
8 enabled and from the same eye position from which the first rendering pass is
9 taken for generating additional shadow maps and for collecting additional
10 colored-transparency information relating to additional depth layers;

- 11 (b) storing the collected colored-transparency information in memory; and
- 12 (c) blending the colored-transparency information from the depth layers.

1 29. A computer program product for transparency rendering in a graphics
2 pipeline, comprising:

3 (a) computer code for collecting colored-transparency information from a
4 plurality of depth layers in a scene to be rendered by:

5 (i) executing a first rendering pass for generating a shadow map and for
6 collecting first colored-transparency information relating to a first depth
7 layer, and

8 (ii) executing additional rendering passes with a shadow mapping feature
9 enabled and from the same eye position from which the first rendering pass is
10 taken for generating additional shadow maps and for collecting additional
11 colored-transparency information relating to additional depth layers;

12 (b) computer code for storing the collected colored-transparency information in
13 memory; and

14 (c) computer code for blending the colored-transparency information from the
15 depth layers.